



“Proyecto Academia Musical JACQUIN”

Taller arquitectura de software

Aprendiz:

Manuel Prudencio Pertuz Pérez

Servicio Nacional de Enseñanza – SENA

Centro Minero Regional Boyacá.

2977343 – Análisis Y Desarrollo de Software.

Instructor:

Luis Edilberto Díaz Sandoval

Julio - 2025.



Tabla de contenido

Introducción	3
1. ¿Qué entiende por arquitectura de software?	4
2. ¿Cuál es su función?	4
3. ¿Cómo se elabora la arquitectura de software?	4
4. ¿Cómo lograr una buena arquitectura?	5
5. ¿Cuáles son los elementos de diseño de una arquitectura de software?	5
Conclusiones	6
Bibliografía	6



Introducción

La arquitectura de software es un pilar fundamental en el desarrollo de sistemas de información modernos. Esta permite establecer un marco organizado para construir, mantener y escalar aplicaciones de forma eficiente. En este taller se abordan conceptos esenciales que permitirán al aprendiz comprender la importancia, estructura y buenas prácticas relacionadas con la arquitectura de software, especialmente aplicadas a proyectos reales como el desarrollo del sistema para *Jacquin Academia Musical*.

1. ¿Qué entiende por arquitectura de software?

La arquitectura de software es la estructura fundamental de un sistema, definida a través de sus componentes, sus relaciones y las reglas que gobiernan su diseño y evolución. Es como el plano de construcción de un edificio, pero aplicado al desarrollo de software.

2. ¿Cuál es su función?

Las funciones principales de la arquitectura de software son:

- Organizar los componentes del sistema para facilitar su desarrollo y mantenimiento.
- Establecer un marco que permita tomar decisiones técnicas fundamentadas.
- Mejorar la calidad del software, garantizando rendimiento, escalabilidad, seguridad y reutilización.
- Servir como guía para los desarrolladores durante todo el ciclo de vida del sistema.

3. ¿Cómo se elabora la arquitectura de software?

El proceso de elaboración de una arquitectura de software incluye:

- **Análisis de requisitos:** comprender lo que el sistema debe hacer y sus restricciones.
- **Selección de patrones de diseño:** identificar estructuras probadas como MVC, capas, cliente-servidor, entre otros.
- **Definición de vistas:**
 - Vista lógica (diagramas de clases, componentes, paquetes).
 - Vista física (diagrama de despliegue).

- **Documentación:** describir claramente los módulos, relaciones, tecnologías utilizadas y sus interacciones.
- **Validación:** verificar que la arquitectura cumple con los requisitos funcionales y no funcionales.

4. ¿Cómo lograr una buena arquitectura?

Para lograr una arquitectura sólida se deben aplicar las siguientes buenas prácticas:

- **Seguir principios SOLID** para un diseño orientado a objetos.
- Usar **patrones de diseño adecuados** (fachada, estrategia, repositorio, etc.).
- Fomentar la **modularidad** y el bajo acoplamiento.
- Aplicar **seguridad desde el diseño**.
- Considerar desde el inicio los atributos de calidad: rendimiento, mantenibilidad, disponibilidad.
- Elegir herramientas que permitan mantener un **flujo de desarrollo eficiente** (IDE, control de versiones, CI/CD, etc.).

5. ¿Cuáles son los elementos de diseño de una arquitectura de software?

Los principales elementos de diseño son:

- **Componentes:** módulos funcionales del sistema (clases, servicios, bibliotecas).
- **Conectores:** cómo se comunican los componentes (APIs, eventos, llamadas, sockets).
- **Interfaces:** puntos de entrada o exposición entre módulos o sistemas externos.
- **Configuraciones:** estructura completa de relaciones entre componentes.
- **Estilos arquitectónicos:** como MVC, microservicios, capas, monolitos.

- **Patrones de diseño:** soluciones reutilizables para problemas comunes de estructura y comportamiento.

Conclusiones

La arquitectura de software es el corazón técnico de cualquier sistema informático. Su correcta planificación y aplicación mejora la calidad, reduce riesgos y permite construir software sostenible en el tiempo. Este taller fortalece las bases para aplicar patrones y estructuras adecuadas en proyectos reales, como el de la Jacquin Academia Musical.

Bibliografía

- Bass, L., Clements, P., & Kazman, R. (2012). *Software Architecture in Practice* (3rd ed.). Addison-Wesley. <https://doi.org/10.5555/2485571>
- Pressman, R. S. (2010). *Ingeniería del software: un enfoque práctico*. McGraw-Hill.
- SENA (2025). *Material formativo del programa Tecnólogo en Análisis y Desarrollo de Software*.
- Fowler, M. (2003). *Patterns of Enterprise Application Architecture*. Addison-Wesley. <https://doi.org/10.5555/579660>